REMARKS

Claims 1-10 and 12-32 are pending in the present application, with claims 1, 12, 22, 26, and 30 being the independent claims. In the Office Action, dated January 11, 2006, claims 30-32 stand rejected under 35 U.S.C. § 101 and claims 1-10 and 11-32 stand rejected under 35 U.S.C. § 102(e) as allegedly anticipated by U.S. Patent No. 6,308,274 (Swift).

Rejection of Claims 30-32 under 35 U.S.C. § 101

Regarding claims 30-32, Applicants previously amended the claims to highlight certainly functional aspects of the recited data structures. The Official Action maintains that such data structures are not within the province of 35 U.S.C. § 101. In this regard, Applicants respectfully request consideration of the seminal case on data structure claims: *In re Lowry*, 32 F.3d 1579, 1584 (Fed. Cir. 1994) in which a similar discussion to the present case was presented concerning Lowry's data structure claims – namely that Lowry's data structure claims were mere arrangement of information in memory, similar to mere "printed material."

As the Court made very clear in that case, however, in affirming the patentability of Lowry's data structure claims, "More than mere abstraction, data structures are specific electrical or magnetic structural elements in a memory. According to Lowry, the data structures provide tangible benefits: data stored in accordance with the claimed data structures are more easily accessed, stored, and erased. Lowry further notes that, unlike prior art data structures, Lowry's data structures simultaneously represent complex data accurately and enable powerful nested operations." Persuaded that Lowry's data structures were more than a mere abstraction of information stored in memory, the Federal Circuit confirmed the patentability of those claims.

Similar to the Lowry claims, the recitations of claims 30-32 are not merely pieces of data. The data structures of claims 30-32 represent more than mere arrangement of information because, for instance, as described in the Summary of the invention, "applications may define and implement business rules that can be expressed in terms of run-time operations and dynamic data. Because this invention places no restrictions on the policy language, an application has substantial flexibility in the definition and implementation of custom authorization policy, while at the same time providing standard APIs and ACL definitions for such dynamic data and policy."

The standardization of ACL definitions and corresponding structure, as defined by claims 30-32, is thus far more than the mere arrangement of information, but rather such data structures enable applications to rely on standard APIs and a standard data structure for implementing dynamic authorization policy. Applicants' data structures thus *do* impart functionality when employed as a computer component, and such functionality has never before been provided in a computing system data structure.

Reconsideration and withdrawal of the rejection based on 35 U.S.C. § 101 is thus respectfully requested.

Rejection of Claims 10 and 12-29 based on Swift

Swift describes that prior art access systems based on access tokens did not have a way of reducing privilege for a user where increased privilege is unnecessary. To solve that problem, Swift provides a mechanism to enforce "least privilege," or in some way reduced access, via restricted access tokens. Restricted tokens enable a security mechanism to determine whether a process has access to a resource based on a modified, restricted version

of an existing access token. The restricted token is based on an existing token, and has less access than the existing token.

The restricted token is created by changing an attribute of one or more security identifiers that allow access in the parent token to a setting that denies access in the restricted token. Similarly, the restricted token can be created by removing a privilege from the restricted token that are present in the parent token. In addition, restricted security identifiers may be placed in a restricted token.

While the restricted token based system of Swift is made more versatile by allowing the creation of restricted access tokens, Swift remains an example of a system that enforces static access policy. The point is that every time a token, e.g., a restricted token, of Swift is evaluated to determine whether a particular task can be performed by a user, or whether an application may access a particular object, *the token or restricted token is evaluated the same way*. Any such system enforces a static access policy, as described in the background section of Applicants' specification (See pages 1-2, first three paragraphs of background).

Still further, Swift implicates static data as well. The restricted tokens of Swift, for instance, include no dynamic data to be evaluated at run-time. Rather, at run-time, the Swift tokens are evaluated the same way every time. In this sense, introducing restricted tokens (based on a parent token) makes the system of Swift no less static than the system Swift modifies.

This distinction between static access policy and data and dynamic access policy and data is found in each of the independent claims, as emphasized below in bold:

1. A method for dynamically managing access to a resource in a computer system, the system having a client thereof making an access request for the resource, the method comprising:

determining, via an application programming interface, based upon dynamic data and first dynamic policy whether a client authorization context is to be updated, wherein said first dynamic policy is tailored to an application through which the resource is accessed;

identifying an access control entry as a callback access control entry; and

in response to identifying the access control entry as a callback access control entry, evaluating, via said application programming interface, based upon dynamic data and second dynamic policy whether said callback access control entry bears on said access request, wherein said second dynamic policy is tailored to said application.

12. A tangible computer readable medium having computer executable instructions stored thereon for carrying out a method for dynamically updating a client authorization context in a computer system, the method comprising:

computing a client authorization context after the request for the resource is received from the client;

determining, via an application programming interface, based upon dynamic data and dynamic policy whether said client authorization context is to be updated, wherein said dynamic policy is tailored to an application through which the resource is accessed; and

updating said client authorization context according to said determination.

22. A tangible computer readable medium bearing computer executable instruction for performing a method of dynamically managing access to a resource in a computer system, the system having a client thereof making an access request for the resource, the method comprising:

comparing the authorization context of the client to at least one access control entry of an access control list;

identifying an access control entry as a callback access control entry; and

in response to identifying the access control entry as a callback access control entry, determining, via an application programming interface, based upon dynamic data and dynamic policy whether said callback access control entry bears on said access request, wherein said dynamic policy is tailored to said application.

30. A data structure stored on a tangible computer readable medium for use in connection with dynamic access check determinations for an application in a computer system, the data structure comprising:

an identifier for identifying the data structure as a callback access control entry; and

dynamic authorization policy data in a format tailored to the application to handle access requests.

Since the system of Swift is based on static policy and data, as described in detail above, Swift cannot be said to teach or suggest at least the above-bolded elements of claims 1, 12, 22 and 30. Claims 2-10, 13-21, 23-29 and 31-32 depend from claims 1, 12, 22 and 30,

respectively, and are believed allowable for the same reasons. Reconsideration and withdrawal of the rejection based on Swift is respectfully requested.

CONCLUSION

Applicants believe that the present Amendment is responsive to each of the points raised by the Examiner in the Official action, and submits that Claims 1-10 and 12-32 of the application are in condition for allowance. Favorable consideration and passage to issue of the application at the Examiner's earliest convenience is earnestly solicited.

Date: March 10, 2006

Thomas E. Watson
Registration No. 43,243

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439